

Enviando archivos por cURL



En el artículo anterior hemos hecho el trabajo "pesado" de configurar nuestra conexión cURL, salvo un par de detalles que dejamos, específicamente, para este artículo. Aquí vamos a ver como enviar archivos por cURL a un servidor remoto, de tipo FTPS, con protocolo FTP implícito sobre TLS (el caso más rebuscado de configurar). De este modo, seremos capaces de programar cualquier envío por cURL.

ATENCIÓN. Recuerda que PHP te ofrece otros tipos de conexión para transferencia de archivos, como puedan ser `ssh2_connect()` o `ftp_connect()`, que no recurren a cURL. Sin embargo, en casos como el propuesto, donde usamos FTP implícito sobre TLS, ninguna de estas alternativas funciona, por lo que cURL es, en muchos casos, la única solución aceptable y, en última instancia, cURL siempre funciona, sea cual sea el protocolo, si está correctamente configurado.

USANDO cURL PARA ENVÍO

Ya casi tenemos todo listo para hacer el envío de un fichero a un servidor remoto por cURL. Empecemos viendo lo que ya tenemos declarado y explicado:

```
<?php
// Valores iniciales
$protocol = 'ftps://'; // El protocolo de comunicación
$server = '245.245.245.245'; // IP del servidor remoto
$port = 990; // Puerto de datos del servidor remoto
$path = ""; // Ruta del servidor donde leeremos o escribiremos
$passive = true; // Un servidor pasivo
$username = 'usuario'; // El nombre de usuario con el que conectar al servidor remoto
$password = 'password'; // La contraseña de acceso al servidor remoto
// Establecemos las opciones básicas
$options = [
    CURLOPT_USERPWD      => $username.':'.$password,
    CURLOPT_SSL_VERIFYPEER => false,
    CURLOPT_SSL_VERIFYHOST => false,
    CURLOPT_FTP_SSL      => CURLFTPSSL_ALL,
    CURLOPT_FTPSSLAUTH   => CURLFTPAUTH_DEFAULT,
    CURLOPT_UPLOAD       => true,
    CURLOPT_PORT         => $port,
    CURLOPT_TIMEOUT      => 30,
];
// Creamos el manejador de una conexión cURL, y le asignamos las opciones que ya tenemos definidas
$curl_handle = curl_init();
foreach ($options as $option => $value) {
    curl_setopt($curl_handle, $option, $value);
}
```

El siguiente paso es establecer la ruta del directorio del archivo remoto donde se depositará el contenido que enviemos por cURL. Este directorio deberá existir en el servidor remoto, ya que cURL no lo crea, sólo deposita el contenido. Si no existe, no se depositará el contenido.

Luego establecemos el nombre del fichero con el que se grabarán los contenidos en el servidor remoto.

Con el protocolo, el servidor, la ruta y el nombre del archivo en remoto, crearemos una URL que asignaremos a la opción `CURLOPT_URL` del manejador cURL que habíamos creado con `curl_init()`.

Todo esto queda así:

```
$remote_path = 'subidas/'; // La ruta base en la que vamos a trabajar en el servidor remoto.
```

```
$dest_file_name = 'remote_test.txt'; // El nombre con el que queremos grabar nuestros contenidos en el servidor remoto.
```

```
$url = $protocol.'/'.$server.'/$remote_path.$dest_file_name; // URL final del archivo en el servidor remoto.
```

```
curl_setopt($curl_handle, CURLOPT_URL, $url); // Establecemos la URL final para el archivo en el servidor remoto en cURL.
```

El siguiente paso es abrir el fichero local, cuyo contenido queremos enviar al servidor remoto, mediante un manejador para lectura.

Después tenemos que asignar ese manejador de lectura a la opción `CURLOPT_INFILE` del manejador cURL.

Después recurrimos al método `curl_exec()`. Este es propio de cURL y es el que se encarga de ejecutar la transferencia.

Lo último que nos queda ya es cerrar los dos manejadores: el de cURL y el del fichero que hemos enviado al servidor remoto.

Todo esto nos queda como ves a continuación:

```
$file_handle = fopen('local_test.txt', 'r');
```

```
curl_setopt($curl_handle, CURLOPT_INFILE, $file_handle);
```

```
curl_exec($curl_handle);
```

```
fclose($file_handle);
```

```
curl_close($curl_handle);
```

CONCLUYENDO

Con esto ya tenemos todo lo necesario para realizar un envío a un servidor remoto desde el servidor donde está nuestra aplicación, mediante cURL. En el próximo artículo veremos otro modo de hacer subidas por cURL.